Quality Assurance Guidelines

Expertise with security standards	→	Cross-platform experience	→	Understand data & workflows	→	Exposure to modern testing tools	→	Automation

Software Release Process _ Quality Assurance

Quality Assurance Gatekeeping

QA acts as a gatekeeper, ensuring that only thoroughly tested and validated software reaches end-users. By implementing quality assurance measures, such as testing, code reviews, and performance checks, software teams can mitigate the chances of releasing a flawed or unstable product. This, in turn, enhances the overall user experience and builds trust among customers. Testing consists of several stages, each serving a specific purpose.

Analyze requirements

Use the information gained during requirements analysis to plan the test.

Evaluate all aspects

Evaluate the product for usability, design, and security.

Plan the QA life cycle

Plan the entire QA life cycle for the software.

Test

Keep your solutions in-line with the end user expectations. Quality assurance engineers carry out functional, integration, and regression testing to detect and rectify any issues. This step is crucial in guaranteeing a stable and reliable software product.

Use comprehensive test data

Test data should be accurate and comprehensive to ensure that defects are detected.

Use regression testing

Regression testing ensures that code changes don't break existing functionality.

Use continuous integration and deployment.

Continuous integration (CI) and continuous deployment (CD) can help ensure smooth software releases.

Communicate

Maintain healthy communication between PMO-Dev and QA.

Report bugs effectively

Detailed report, properly filed and structured, clear and concise, contains meaningful examples.